TIME AND MEMORY REQUIREMENTS FOR SOLVING LINEAR SYSTEMS

BY

MARIA AURORA MORGANA

STAN-CS-71-212

MARCH, 1971

COMPUTER SCIENCE DEPARTMENT

School of Humanities and Sciences

STANFORD UNIVERSITY

TIME AND MEMORY REQUIREMENTS FOR SOLVING LINEAR SYSTEMS

Maria Aurora Morgana

Stanford University
Stanford, California

Abstract

The Computer Science Department program library contains a number of ALGOL W procedures and FORTRAN subroutines which can be used to solve systems of linear equations.

This report describes the results of tests to determine the amount of time and memory required to solve systems of various orders.

This research was supported by the Atomic Energy Commission (AT(04-3) 326-PA30), the National Science Foundation (GJ-408), and the Office of Naval Research (N00014-67-A-0112-0029 NR 044-211).

TIME AND MEMORY REQUIREMENTS FOR SOLVING LINEAR SYSTEMS

Maria Aurora Morgana

Stanford University
Stanford, California

Introduction

The Computer Science Department program library contains a number of ALGOL W procedures and FORTRAN subroutines which can be used to solve systems of linear equations. These include:

ALGOL W		FORTRAN	
single prec.	double prec.	single prec.	double prec.
DECOMPOSE	LONGDECOMPOSE	LINSY1	LINSY2
SOLVE	LONGSOLVE	DECMP1	DECMP2
IMPROVE	LONGIMPROVE	SOLVE1	SOLVE2
SINGULAR	LONGSINGULAR	IMPRV;	IMPRV2
	IP2		

During the winter and autumn of 1970 I tested these programs to determine the amount of time and memory required to solve systems of various orders. This report describes the results of these tests.

Preliminaries

The linear system $Ax = b$ used in all the tests was composed of a matrix A of elements with random values between -.5 and .5 generated using the programs RANDOM in ALGOL W (with initial value=1) and RANDK in FORTRAN (with initial value=1), and a right-hand-side vector b formed

by summing the columns of A. Since both routines return random numbers with values between 0.0 and 1.0, each value was subtracted from 0.5 to give random numbers in the desired range. A 100 x 100 matrix was generated row after row. Systems of order less than 100 were formed using the principal minors of this larger matrix. (The values of a(1,1) and a(100,100) were .1423882 and -.1315178, respectively.)

Execution times were computed using the function PCLOCK in FORTRAN programs compiled by IBM's compiler, and the integer procedure TIME in ALGOL W programs. No similar routine is available to FORTRAN programs compiled with the WATFIV compiler.

Other factors considered besides language and precision of data were:

1. FORTRAN compilers: Two compilers, WATFIV and IBM's FORTRAN IV H-level compiler, are available for FORTRAN programs.
2. Compiler options: IBM's FORTRAN compiler allows various levels of code optimization. OPT=0 and OPT=2 were examined. In addition, the WATFIV compiler allows the option of checking for uninitialized variables, and the ALGOL W compiler the option of checking array subscripts to see whether they lie within their proper bounds.
3. Available memory: Jobs were run with QUICK, STANDARD, and OVERNIGHT priorities to determine the maximum order of system which can be solved in the partition sizes available to the user.

These tests were run using the level 13 release of IBM's operating system OS/360, level 17 of the FORTRAN IV compiler, the 21 September, 1969,

version of the ALGOL W compiler, and the version 1, level 1, January, 1970, WATFIV compiler.

Results

Time comparisons between different compilers for a system of 50 equations in single and double precision follow, where

TS = time to solve the system initially,
TI = time to improve the solution
TT = total time,

and times are given in hundredths of a second.

SINGLE PRECISION:

	TS	TI	TT	Total cost at $9/min.
FORTRAN H				
OPT = 2	63	18	81	$.12
OPT = 0	280	52	332	.50
ALGOL W				
without sub. check	393	52	445	.66
with sub. check	715	78	793	1.18
WATFIV*			1274	1.91

DOUBLE PRECISION:

	TS	TI	TT	Total cost at $9/min
FORTRAN H				
OPT = 2	82	172	254	.38
OPT = 0	304	234	538	.80
ALGOL W				
without sub. check	400	322	722	1.08
with sub. check	720	432	1152	1.72
WATFIV*			2206	3.31

* The times given for programs compiled with the WATFIV compiler are total run times, including the time required to generate the system, print out the results, etc. Comparable times for ALGOL W programs are 668 and 805. Also, it was found that the difference in run times for programs in which checking for initialized variables was performed and those in which it wasn't, was negligible.

If one takes into account total execution time, including compile, run, and system overhead times, then it is advisable to use FORTRAN programs compiled with IBM's compiler with OPT=2 for a single system of order 70 and above since the run times are so much shorter than those of programs compiled with the WATFIV compiler. For a single system of order less than 70 it is advantageous to use the latter because of shorter compile and system overhead times.

It should be pointed out that the increased time necessary to solve a system of equations in double precision is due to the use of the subroutines DPPUT and IPTOTL in FORTRAN and the procedure IP2 in ALGOL W. These routines are used to form inner products in double long precision in order to improve the initial solution. Note that the times required to obtain an initial solution are about the same in both cases. In single precision the time necessary to improve the solution is only about 10% of the total time, while in double precision this percentage increases noticeably.

The times to solve the system and the total times required for different values of N, using the FORTRAN programs compiled with IBM's compiler with OPT=2, and using the ALGOL W procedures without subscript checking, were fitted with a cubic polynomial using a least squares criterion. Similarly, the times to improve were fitted with a quadratic polynomial. These various times may then be computed for different values of N, using the formula

$$\text{time}(n) = \sum_{i=0}^{3} a(i)*(n/10)**i,$$

where the values of the coefficients are given below. TS, TI, and TT are as defined before, and times are again given in hundredths of a second.

FORTRAN Times

	Single precision			Double precision		
N	TS	TI	TT	TS	TI	TT
10	1	1	2	1	4	5
20	5	3	8	6	15	21
30	15	7	22	20	62	82
40	34	12	46	43	109	152
50	63	18	81	82	172	254
60	107	26	133	139	240	379
70	167	37	204	216	324	540
80	247	46	293	317	431	748
90	348	58	406	451	532	983
100	469	70	539	645	691	1336

Coefficients: single precision

	a(0)	a(1)	a(2)	a(3)
time to solve	2.10	-1.96	0.83	0.40
time to improve	-0.67	0.62	0.65	
total time	3.23	-2.94	1.83	0.38

Coefficients: double precision

time to solve	-14.00	14.85	-3.45	0.85
time to improve	- 2.00	-1.00	6.91	
total time	-36.33	31.79	-0.45	1.09

ALGOL W Times

	Single precision			Double precision		
N	TS	TI	TT	TS	TI	TT
10	5	2	7	5	17	22
20	30	8	38	30	55	85
30	92	18	110	93	122	215
40	207	33	240	210	207	417
50	393	52	445	400	322	722
60	663	73	736	690	453	1143
70	1046	97	1143	1070	617	1687
80	1535	128	1663	1580	792	2372
90	2173	160	2333	2231	1002	3233
100	2965	195	3160	3045	1230	4275

Coefficients: single precision

	a(0)	a(1)	a(2)	a(3)
time to solve	-1.93	2.67	1.11	2.83
time to improve	-2.03	1.52	1.83	
total time	-2.17	2.53	3.28	2.81

Coefficients: double precision

time to solve	0.73	-0.87	2.00	2.85
time to improve	13.07	-1.89	12.16	
total time	0.43	3.79	13.92	2.84

The order of the largest solvable system is limited by the size of the partition in which the program is executed. (In FORTRAN, the maximum order is fixed at 100 by the routines but can be increased by changing the dimension of the vectors SCALES in DECMP1 and DECMP2, and R and DX in IMPRV1 and IMPRV2.)

The amount of storage required by the minimum number of arrays necessary to solve a system of order N is

(2*N**2 + 6*N) * 4 bytes in single precision

and

(2*N**2 + 6*N) * 8 bytes in double precision.

Using this minimum storage requirement, the orders of the largest systems which can be solved using the FORTRAN subroutines compiled with IBM's compiler and the ALGOL W procedures are approximately

FORTRAN

	single prec.	double prec.
standard partition	200	140
overnight partition	245	170

ALGOL W

	single prec.	double prec.
quick partition	105	75
standard partition	195	140
overnight partition	250	175

Unclassified

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1 ORIGINATING ACTIVITY *(Corporate author)* Stanford University	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3 REPORT TITLE

TIME AND MEMORY REQUIREMENTS FOR SOLVING LINEAR SYSTEMS

4 DESCRIPTIVE NOTES *(Type of report and inclusive dates)*
Technical Report, March 1971

5 AUTHOR(S) *(First name, middle initial, last name)*

Maria Aurora Morgana

6 REPORT DATE March 1971	7a. TOTAL NO. OF PAGES 7	7b. NO. OF REFS 0

8a. CONTRACT OR GRANT NO ONR N00014-67-A-0112-0029 NR 044-211, b. PROJECT NO. AEC AT (04-3) 326-PA30), NSF GJ-408. c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) STAN-CS-71-212 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)* None

10 DISTRIBUTION STATEMENT

Releasable without limitations on dissemination.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Mathematics Program Office of Naval Research Arlington, Virginia 22217

13. ABSTRACT

The Computer Science Department program library contains a number of ALGOL W procedures and FORTRAN subroutines which can be used to solve systems of linear equations.

This report describes the results of tests to determine the amount of time and memory required to solve systems of various orders.

DD FORM 1 NOV 68 1473 (PAGE 1)

S/N 0101-807-6801

Unclassified

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Computing times Storage requirements Matrix computations						